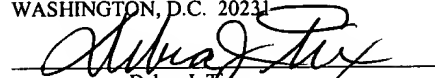


PATENT
5053-36000

"EXPRESS MAIL" MAILING
LABEL NUMBER :
EL493675560US
DATE OF DEPOSIT JUNE 23, 2000
I HEREBY CERTIFY THAT THIS
PAPER OR FEE IS BEING
DEPOSITED WITH THE UNITED
STATES POSTAL SERVICE
"EXPRESS MAIL POST OFFICE TO
ADDRESSEE" SERVICE UNDER 37
C.F.R. § 1.10 ON THE DATE
INDICATED ABOVE AND IS
ADDRESSED TO THE
COMMISSIONER OF PATENTS
AND TRADEMARKS,
WASHINGTON, D.C. 20231


Debra J. Tix

**INTERNET-ENABLED SYSTEM AND METHOD
FOR ASSESSING DAMAGES**

By:

Scott Lorenz

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to computer software, and more particularly to insurance claim processing software.

2. Description of the Related Art

Insurance companies have been processing and settling claims associated with bodily injury for a long time. The task of evaluating, analyzing or estimating the amount of damage associated with one or more types of bodily injuries, especially trauma-induced bodily injuries, can be very complex. Complexity in the evaluation process often arises out of the fact that concurrent expertise in legal, medical and insurance fields is often required to arrive at a particular decision involving a bodily injury claim.

Several factors can affect the estimated amount of the claim associated with a bodily injury. Every accident is different and every injury is unique. Arriving at a customized evaluation of a bodily injury claim, which is unique for a specific accident, injury, etc. is desirable. Applying across-the-board standards may tend to result in an inequitable solution for one or more parties involved. External environmental factors, such as the experience level of a claims adjuster, record of accomplishment of the legal professionals, post-injury quality of life for the injured party, etc., all can affect the valuation of a claim.

During the past several years, many insurance companies have been using computer-based and knowledge-based claim-processing systems to process, evaluate, analyze and estimate thousands of claims in a fair and consistent manner. A knowledge-based claim-processing system includes an expert system which utilizes and builds a knowledge base to assist the user in decision making. It may allow the insurance companies to define new rules and/or use previously defined rules, in real-time. The

business rules are generally written by industry experts to evaluate legal, medical, insurance conditions before arriving at a valuation of a claim.

In knowledge-based systems, to estimate a claim for bodily injury, the user may enter inputs on a display screen and step through a series of displays or screens to complete the data input process. This process may be referred to as a consultation session. The knowledge-based claim processing system may then utilize the user-provided inputs to generate a consultation report. The consultation report may include, for example, an estimate of a value of an insurance claim.

In the past, such knowledge-based systems for estimating the value of an insurance claim have been limited to traditional computing architectures such as mainframes and stand-alone personal computers. Therefore, it was necessary to install and maintain client software as well as server software for these knowledge-based systems in particular physical locations. With the growth of the Internet, however, many personal computers may now be granted client access to servers distributed all over the world.

It may therefore be desirable to develop a knowledge-based system for insurance claim processing which is configured to be accessed over the Internet. It may also be desirable to develop a knowledge-based system for insurance claim processing which is configured to be accessed through a web browser. It may be desirable to develop new business models for providing insurance companies with access to a web-based or internet-based insurance claim processing server.

SUMMARY OF THE INVENTION

The present invention provides various embodiments of an Internet-enabled method and system for processing insurance claims. The system may include an insurance claim processing server which may include a first CPU and a first memory coupled to the first CPU. The first memory may store a first set of program instructions which are executable by the first CPU to estimate a value of an insurance claim as a function of insurance claim assessment data entered by a user during an insurance claim consultation session. The insurance claim may include a bodily injury claim, and the estimate of the value of the insurance claim may include an estimate of bodily injury general damages. The insurance claim assessment data may include one or more bodily injuries and one or more treatments of the bodily injuries. The first set of program instructions may include a rules engine and a web server. The first set of program instructions may be further executable by the first CPU to generate and send to a client computer system a plurality of web pages comprising insurance claim assessment questions.

The system may also include a client computer system which may include a second CPU and a second memory coupled to the second CPU. The client computer system may be coupled to the insurance claim processing server through a network. The network may include the Internet, and the insurance claim processing server and the client computer system may therefore be operable to communicate over the network via TCP/IP. The second memory may store a second set of program instructions which are executable by the second CPU to receive the insurance claim assessment data entered by the user and send the insurance claim assessment data across the network to the insurance claim processing server. The second set of program instructions may include a web browser. The second set of program instructions may be further executable by the second CPU to display the web pages comprising the insurance claim assessment questions during the insurance claim consultation session.

[illegible]

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a network diagram of an illustrative distributed computing environment which is suitable for implementing various embodiments;

Figure 2a is an illustration of an insurance claims processing server computer architecture according to one embodiment;

Figure 2b is an illustration of an insurance claims processing client computer architecture according to one embodiment;

Figure 3a is an illustration of an insurance claims processing server software architecture for a single client according to one embodiment;

Figure 3b is an illustration of an insurance claims processing server software architecture for multiple clients according to one embodiment;

Figure 4 is an illustration of adapter software between a rules engine and a web server according to one embodiment;

Figure 5 illustrates the transmission of data between a web server and a web browser according to one embodiment;

Figure 6 illustrates an example of a browser-based user interface for the insurance claims processing system according to one embodiment;

Figure 7 is a flowchart illustrating a method of developing a web-based insurance claims processing system according to one embodiment;

Figure 8 is a flowchart illustrating a method of hosting a web-based insurance claims processing server with various pricing models according to one embodiment;

Figure 9 is a flowchart illustrating a method of using a reset button provided by a web-based interface to a web-based insurance claims processing server according to one embodiment.

While the invention is susceptible to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and will herein be described in detail. It should be understood, however, that the drawings and detailed description thereto are not intended to limit the invention to the particular form disclosed, but on the contrary, the intention is to cover all modifications, equivalents, and

alternatives falling within the spirit and scope of the present invention as defined by the appended claims.

DETAILED DESCRIPTION OF SEVERAL EMBODIMENTS

Figure 1: A Distributed Computing Environment

Figure 1 is a network diagram of an illustrative distributed computing environment which is suitable for implementing various embodiments. The distributed computing environment may include various server systems 70 and client systems 80 connected by a network 55. Other networkable devices such as printers 90 may also be connected to the network 55. The servers 70, clients 80, and other devices may be geographically dispersed. A single computer system may serve as both a server and client.

The network 55 may be a local area network or wide area network, and may include communications links including, but not limited to: Ethernet, token ring, Internet, satellite, wireless, telephone, cable, DSL, and other suitable pathways. As used herein, "the Internet" includes one or more substantially global networks which are generally accessible by the public (i.e., they are not proprietary or not largely characterized by controlled access). Various sources of data on the Internet may be accessed through protocols such as HTTP (HyperText Transport Protocol), HTTPS (Secure HyperText Transport Protocol), FTP (File Transfer Protocol), Telnet, NNTP (Network News Transport Protocol), SMTP (Simple Mail Transfer Protocol), and other suitable protocols. Transmission of data over the Internet is typically achieved through the use of TCP/IP (Transmission Control Protocol/Internet Protocol) packets.

Figures 2a and 2b: An Insurance Claims Processing Server Computer Architecture and an Insurance Claims Processing Client Computer Architecture

Figure 2a is an illustration of an insurance claims processing server computer architecture according to one embodiment. Figure 2b is an illustration of an insurance claims processing client computer architecture according to one embodiment. The insurance claims processing server 70 may include a computer system 20a with a memory 30a. The insurance claims processing client 80 may include a computer system

20b with a memory 30b. The term "computer system" as used herein generally includes the hardware and software components that in combination allow the execution of computer programs. The computer programs may be implemented in software, hardware, or a combination of software and hardware. A computer system's hardware generally includes a processor, memory media, and Input/Output (I/O) devices. As used herein, the term "processor" generally describes the logic circuitry that responds to and processes the basic instructions that operate a computer system. The term "CPU" is used synonymously with "processor" herein. The term "memory" is used synonymously with "memory medium" herein. The term "memory medium" is intended to include an installation medium, e.g., a CD-ROM, or floppy disks, a volatile computer system memory such as DRAM, SRAM, EDO RAM, Rambus RAM, etc., or a non-volatile memory such as optical storage or a magnetic medium, e.g., a hard drive. The memory medium may comprise other types of memory as well, or combinations thereof. In addition, the memory medium may be located in a first computer in which the programs are executed, or may be located in a second different computer that connects to the first computer over a network 55. In the latter instance, the second computer provides the program instructions to the first computer for execution. In addition, the computer system may take various forms, including a personal computer system, mainframe computer system, workstation, network appliance, Internet appliance, personal digital assistant (PDA), television system or other device. In general, the term "computer system" can be broadly defined to encompass any device having a processor that executes instructions from a memory medium.

The memory medium preferably stores a software program or programs for processing insurance claims as described herein. The software program(s) may be implemented in any of various ways, including procedure-based techniques, component-based techniques, and/or object-oriented techniques, among others. For example, the software programs may be implemented using a rule-based development tool such as PLATINUM Aion™ available from Computer Associates International, Inc. In one embodiment, PLATINUM Aion™ may combine business rule and object-oriented technologies to create and maintain complex, knowledge-intensive applications.

Software developed with PLATINUM Aion™ may employ an Aion™ programming language for automation of processes which may use hundreds or thousands of business rules from a knowledge base. An Aion™ inference engine may automatically determine which rules to execute, when, and in what order. In various other embodiments, the software program may be implemented using other technologies, languages, or methodologies, as desired. A CPU, such as the host CPU, executing code and data from the memory medium includes a means for creating and executing the software program or programs according to the methods, flowcharts, and/or block diagrams described below.

Various embodiments further include receiving or storing instructions and/or data implemented in accordance with the description herein upon a carrier medium. Suitable carrier media include memory media or storage media such as magnetic or optical media, e.g., disk or CD-ROM, as well as transmission media or signals such as electrical, electromagnetic, or digital signals, conveyed via a communication medium such as network 55 and/or a wireless link.

A computer system's software generally includes at least one operating system, a specialized software program that manages and provides services to other software programs on the computer system. Software may also include one or more programs to perform various tasks on the computer system and various forms of data to be used by the operating system or other programs on the computer system. The data may include but are not limited to databases, text files, and graphics files. A computer system's software generally is stored in non-volatile memory or on an installation medium. A program may be copied into a volatile memory when running on the computer system. Data may be read into volatile memory as required by a program.

A server may be defined as a computer program that, when executed, provides services to other computer programs (referred to as clients) executing in the same or other computer systems. The computer system on which a server program is executing may also be referred to as a server, though it may contain a number of server and client

programs. In the client/server model, a server is a program that awaits and fulfills requests from client programs in the same or other computer systems.

The insurance claims processing server 70 may further include a display device 50a connected to the computer system 20a and an insurance database 40 residing on an internal or external storage. As used herein, a “database” may include a collection of information from which a computer program may select a desired piece of data. As used herein, an “insurance database” is used as a synonym for a “database” when included in or coupled to an insurance claims processing system. Computer system 20a includes memory 30a configured to store computer programs for execution on the computer system 20a and a central processing unit (or CPU, not shown) configured to execute instructions of computer programs residing on the computer system 20a. Insurance claims processing server software 60 may be stored in the memory 30a. As used herein, an “insurance claims processing program” 60 or “insurance claims processing software” or “insurance claims processing system” may include a software program which is configured to conduct transactions regarding insurance claims, such as by estimating the value of the insurance claims, for example.

The insurance claims processing client 80 may further include a display device 50b connected to the computer system 20b. Computer system 20b includes memory 30b configured to store computer programs for execution on the computer system 20b and a central processing unit (or CPU, not shown) configured to execute instructions of computer programs residing on the computer system 20b. Insurance claims processing client software 68, such as web browser software, may be stored in the memory 30b.

The insurance claims processing server 70 may be connected to network 55. The insurance claims processing server software 60 and insurance database 40 may be distributed among the one or more servers 70 to provide a distributed processing system for insurance claim transactions. In other words, an insurance claim processing transaction being processed by the insurance claim processing system may be routed to any server based upon the workload distribution among servers 70 at the time of the

transaction. Insurance claim processing system servers 70 may be located on a local area network or may be geographically dispersed in a wide area network.

One or more clients 80 may also be connected to network 55. Clients 80 may reside at one or more claim processing units within the insurance company. In a wide area network, clients 80 may be geographically dispersed. Clients 80 may be used to access one or more insurance claim processing system servers 70 and associated insurance databases 40. An insurance claim processing employee may use a client 80 to access the insurance claim processing system and execute insurance transactions. An employee may also use a client 80 to enter insurance claim inputs into the insurance claim processing system. As shown in Figure 1, one or more printers 90 may also be connected to network 55 for printing documents associated with insurance claim transactions.

The insurance claims processing server 70 and client 80 may be used by an Insurance Company for various embodiments of a system and method for processing insurance claims. As used herein, an Insurance Company (IC) includes a business organization that provides insurance products and/or services to customers. More particularly, the insurance products may pertain to providing insurance coverage for accidents and the trauma-induced bodily injuries that may result due to the accident. Examples of trauma-induced bodily injuries may include, but are not limited to: loss of limb(s); bone fractures; head, neck and/or spinal injury, etc.

In one embodiment, on receiving a trauma-induced bodily injury, a customer may file an insurance claim (IC) with his/her insurance organization to cover medical and other accident-related expenses. An IC may utilize a computer-based insurance claim processing system to process insurance claims. In one embodiment, the processing may include estimating a value, including a bodily injury general damages amount, associated with the filed insurance claim.

As used herein, an IC business transaction may be defined as a service of an IC. Examples of business transactions include, but are not limited to: insurance transactions such as filing of claims, payment of claims, application for insurance coverage, and customized benefits, etc. Business transactions may also include services related to customers, insurance providers, employers, insurance agents, investigators, etc.

~~As used herein, an IC insurance claim processing includes a series of instructions executed by a computer system 70 and/or 80 for processing an IC's business transactions. A claim processing system may include one or more processing tasks. A processing task may include a sequence of one or more processing steps or an ordered list or a structured list of one or more processing steps which associated with the business transaction to be processed by the claim processing system. In one embodiment, the sequence of steps may be fixed. In another embodiment the sequence of steps may be established dynamically, in real-time. In one embodiment, the sequence of one or more steps may include an initial step, a final step, one or more intermediary steps, etc. In one embodiment, an IC user may select steps to process an insurance claim in a sequential manner. In another embodiment, the IC user may select steps to process an insurance claim in a random or arbitrary manner. Examples of processing steps may include, but are not limited to: receiving an input from a user of the IC insurance claim processing system 70 and/or 80, reading a value from a database, updating a field in a database, displaying the results of a business transaction on a computer screen, etc.~~

In one embodiment, the insurance claim processing system 70 and/or 80 utilizes object-oriented technology to process insurance claims. In another embodiment, processing of insurance claims may utilize traditional programming languages and databases to achieve the same result. Insurance objects may be defined to represent or model real-world business features of insurance products and services. Examples of insurance objects may include, but are not limited to, objects representing the following: an insurance claim; an accident report; a settlement; an estimated claim; IC service facilities, customers, and employees; business process such as a new insurance application and calculation of a premium; interfaces to external insurance organizations; work tasks such as

calculations, decisions, and assignments; temporal objects such as calendars, schedulers, and timers; and elemental data necessary to accomplish work tasks such as medical costs, risk factors, etc.

An insurance object may be represented on the computer screen 50a and/or 50b by a graphical icon or by a display listing the properties of the insurance object in graphic and/or alphanumeric format. An insurance claim object may be configured to gather and evaluate data for processing a filed insurance claim and to automatically make decisions about the insurance claim. The one or more processing steps associated with the processing of an insurance claim may also be configured as one or more processing step objects. In one embodiment, a display screen, which also may be referred to as a page, may be associated with a processing step. The display screen may also be represented as an object. Each display screen object may include a property to point to a previous display and another property to point to a next display screen. Each property, e.g. the next display pointer on a display screen object, may be changed dynamically by using methods associated with the display screen object. One display screen object may serve as the starting point for processing insurance claims. In one embodiment, the starting point for processing insurance claims may include acquiring an insurance claim identification number from an IC system user.

In one embodiment, during the processing of an insurance claim, a business rule and/or an IC system user input may determine that the insurance claim processing needs the execution of additional steps or tasks to continue the processing of the claim. The IC system user may provide inputs to the insurance claims processing server software 60 at any display screen associated with a step included in a Table of Contents. The insurance claim processing software may dynamically modify the number of steps and/or the sequence of their execution to complete the claim processing transaction. An IC system user working at a client system 20b may then iterate through the claim processing steps and arrive at an estimated value for the insurance claim.

In one embodiment, upon startup, the server software 60 and browser software 68 may provide a graphical user interface to display claims processing related information on display device 50a and/or 50b. It may collect user inputs which are associated with insurance claims and entered by using user input devices 52a and/or 52b. It may process the user inputs, access an insurance database 40, use the contents of the insurance database 40 to estimate the insurance claim, and store it in memory 30a and/or 30b and/or insurance database 40. The browser 68 may display a value of the estimated insurance claim on display screen 50b. A user may view the display of the estimated insurance claim on display screen 50b and may interactively make modifications, additions, and deletions to the estimated insurance claim.

Systems 20a and 20b may also include one or more user input devices 52a and 52b, such as a keyboard, for entering data and commands into the insurance claim program 60. It may also include one or more cursor control devices 54a and 54b such as a mouse for using a cursor to modify an insurance claim viewed on display screen 50a and/or 50b. In response to the updating of the estimated insurance claim, the insurance claim server software 60 may store the updated insurance claim in the insurance database 40.

Figure 3a: An Insurance Claims Processing Server Software Architecture

Figure 3a is an illustration of an insurance claims processing server software 60 architecture for a single client according to one embodiment. The server software 60 may include an insurance processing rules engine 61. As used herein, a “rules engine” may include an expert system which is operable to produce an output as a function of a plurality of rules. A rules engine, in one embodiment, may include an expert computer system which utilizes and builds a knowledge base developed in the form of business rules and/or formulas to assist the user in decision-making. In one embodiment, the rules engine 61 is operable to generate insurance claim assessment questions to be displayed to a user during an insurance claim consultation session. The rules engine 61 may also be operable to estimate a value of an insurance claim as a function of insurance claim

assessment data entered by a user in response to the insurance claim assessment questions. In one embodiment, the insurance claim may include a bodily injury claim, the insurance claim assessment questions may include bodily injury claim assessment questions, the insurance claim assessment data may include bodily injuries and treatments thereof.

In one embodiment, the rules engine 61 is capable of processing rules associated with assessing bodily injury damages claims. A rules engine 61, in one embodiment, comprises an expert computer system which utilizes and builds a knowledge base developed in the form of business rules to assist the user in decision-making. It allows the insurance companies to capture the knowledge base of their experts by defining business rules. Once created, the expertise may be used in processing many transactions, including assessing bodily injury damages claims. The business rules enable claim-processing professionals to be assisted by industry experts to evaluate legal, medical, insurance conditions before arriving at a valuation of an insurance claim.

In various embodiments, the rules engine 61 may be implemented and executed on various computing platforms such as personal computers and mainframes. The rules engine 61 may comprise a rules engine executable file on these platforms. In various embodiments, the rules engine may be accessed through various user interfaces, such as a graphical user interface for a rules engine 61 which is executable on a Microsoft™ Windows™-based server 70. In one embodiment, the rules engine 61 may be developed using a commercial rule-based development tool such as PLATINUM Aion™, which is available from Computer Associates International, Inc. In one embodiment, the rules may be customized to meet the requirements of a particular insurance company.

Business rules, often referred to simply as rules, may include executable computer program instructions. The rules include computer commands or logical instructions to achieve a certain function. For example, rules may guide an assessment or estimate of bodily injury general damages. Each rule, in one embodiment, includes a premise followed by one or more resulting actions. For example, in one embodiment, a business

rule may state 'If patient requires hospitalization after emergency care treatment then the trauma severity level should be classified as major'. In this case, the premise is 'patient requires hospitalization after emergency care treatment'. The resulting action is 'trauma severity level should be classified as major'. In one embodiment, the insurance claim processing server 70 may include several thousand business rules. The rules may be executed or fired, under the control of the insurance claim processing software, based on certain events, user inputs, etc. Only pertinent rules, i.e., a subset of all the available rules, are typically selected and executed for processing a specific bodily injury damages claim. On execution of the plurality of rules which are applicable to a specific bodily injury claim consultation session, the insurance claim processing server software 60 may generate a consultation report which summarizes an assessment and/or estimate of the bodily injuries claim.

The rules may be stored in and retrieved from an insurance database 40. The type of information stored and/or retrieved may include, but not be limited to, business objects, tables, rules, software source code, executable software, etc. In one embodiment, the database may include a relational database. In another embodiment, the database 40 may include an object-oriented database.

In one embodiment, the insurance claims processing server software 60 may include adapter software 62 which may provide access to the rules engine for one or more other computer-based applications or subsystems, such as an internet information server 64. In one embodiment, the adapter software 62 provides an application programming interface (API) to the rules engine 61. The adapter software 62 is discussed in greater detail with reference to Figure 4.

In one embodiment, the insurance claims processing server software 60 may include a web server such as an internet information server (IIS) 64. As used herein, a "web server" includes a system for supplying clients with access to web pages, such as by sending the pages to clients via an appropriate protocol. In one embodiment, a web server may also be operable to generate the web pages dynamically. As used herein, a

“web page” includes a block of information which is configured to be displayed by a web browser 68. As used herein, a “web browser” or “browser software” includes software which is configured to receive and display web pages. Examples of web browsers include Internet Explorer™ available from Microsoft™ Corporation and Netscape Navigator™ available from Netscape Communications Corporation. Typically, a web page is configured to be displayed in a single window in a web browser, wherein the window may be scrolled to view off-screen elements of the web page. Web pages may include various combinations of text, graphics, audio content, video content, and other multimedia content. A web page is often encoded in a language such as HTML (HyperText Markup Language). Web pages may be viewed in a browser on the same computer system on which the server 64 or web pages reside. Web pages may also be transmitted to a client computer system over a network 55, such as via the HyperText Transport Protocol (HTTP) 56. Where the network 55 includes the Internet, the web pages may be transmitted via standard protocols such as TCP/IP.

In one embodiment, the internet information server (IIS) 64 may include a commercial product such as Microsoft™ Internet Information Server available from Microsoft™ Corporation. In one embodiment, the server 64 may include an active server pages (ASP) controller 65 which is operable to generate web pages dynamically. In other words, the web pages delivered by the internet information server 64 may be built in real time by the ASP controller 65 upon a request for a page by a browser 68. Active server pages may include dynamic web pages which are created, for example, by blending HTML and server-side scripting. Active server pages may be dynamically constructed to include insurance claim assessment questions and other user interface elements by starting from a template.

The web server 64 may be configured to generate a plurality of web pages comprising the insurance claim assessment questions. The web browser 68 may then be configured to display the plurality of web pages comprising the insurance claim assessment questions. The web browser 68 may then be configured to receive insurance claim assessment data entered by a user in response to the insurance claim assessment

questions during an insurance claim consultation session and send the insurance claim assessment data to the web server 64. In one embodiment, the web server 64 is further configured to receive the insurance claim assessment data from the web browser 68 and send the insurance claim assessment data to the rules engine 61. The rules engine 61 may be further configured to generate and send the estimate of the value of the insurance claim to the web browser 68 through the web server 64. The web browser 68 may be further configured to display the estimate of the value of the insurance claim received from the rules engine 61 through the web server 68.

In one embodiment, the web server 64 and web browser 68 may be located on separate computer systems which are communicatively coupled through a network 55. In another embodiment, the web server 64 and web browser 68 may be located and executed on a single computer system.

Figure 3b: An Insurance Claims Processing Server Software Architecture

HTTP is considered to be a stateless internet access protocol. In other words, each request from a web browser 68 to a web server 64 is essentially a request-response interaction. Therefore, when a web browser 68 requests a web page, for example, the web server 64 may complete the interaction between the two by sending the page to the browser 68. However, a consultation session conducted by a user through a web browser 68 which communicates with the rules engine 61 may include many successive interactions through the web server 64. It would tend to be inefficient to start a rules engine executable file for each of the many interactions that may take place during a single consultation session.

Therefore, IIS sessions may be used to maintain resources and state for each of a plurality of users. Figure 3b is an illustration of an insurance claims processing server software architecture for multiple clients 68a, 68b, 68c according to one embodiment. The first time a user connects to a suitable web site provided by the server 64, a rules engine may be executed or started for that particular user and then "held" in an IIS

session for that user. Figure 3b illustrates an example including three browsers 68a, 68b, 68c which correspond to and communicate with respective rules engines 61a, 61b, 61c. Each IIS session may include an individual ASP controller 65a, 65b, 65c. Each rules engine 61a, 61b, 61c may therefore be linked to its corresponding ASP controller 65a, 65b, 65c through individual adapter software 62a, 62b, 62c.

Figure 4: Adapter Software

Figure 4 is an illustration of adapter software between a rules engine and a web server according to one embodiment. The adapter software 62 may include one or more components which permit software such as applications or other components to communicate with the rules engine 61. For example, the adapter software may provide methods to start and communicate with a rules engine executable file 61.

As used herein, a component is a software object which includes definitions of method of communication for that software object. Typically, components are implemented according to a component architecture specification such as the Component Object Model (COM) or Distributed Component Object Model (DCOM) promulgated by Microsoft™. The component architecture specification for COM enables applications and components which follow the specification to pass data, commands, and other information back and forth. A COM interface may be said to “wrap” an object, server, or other piece of software if that COM interface defines methods of interaction or communication with that object, server, or piece of software.

In one embodiment, the adapter software 62 may include one or more COM components 63b and a dynamic link library (DLL) 63a. As used herein, a DLL may include a library of executable functions or data that can be used by an application such as a Microsoft™ Windows™-based application. Typically, a DLL provides one or more particular functions, and a program may access those functions by creating either a static or dynamic link to the DLL. A static link remains constant during program execution, while a dynamic link is created by the program as needed. In one embodiment, the DLL

63a may provide a lower-level interface to the functions and methods of the rules engine 61. For example, the DLL 63a may take advantage of published protocols for accessing a rules engine implemented with a commercial system such as PLATINUM Aion™. In one embodiment, the DLL 63a may be provided by the supplier of the commercial system for developing a rules engine.

The COM component(s) 63b may then provide a higher-level interface to the DLL 63a, which in turn may provide an interface to the rules engine 61. In other words, the “business intelligence” may be confined to the rules engine 61 and DLL 63a, and the COM component(s) 63b may expose an interface which permits other pieces of software to convert data, requests, and other parameters to function calls provided by the DLL 63a. In one embodiment, the COM component(s) 63b may include methods including, but not limited to, the following: `setListParameter`, `setSingleParameter`, `getNextMessage`, `lastErrorMessage`, `sendMessage`, `terminateSession`, `transactMessage`, `getListParameter`, `getSingleParameter`, `startServerSession`, and `startRefsysSession`. Appropriate parameters may be defined for each method.

Figure 5: Transmission of Data Between a Web Server and a Web Browser

Figure 5 illustrates the transmission of data between a web server and a web browser according to one embodiment. Each ASP controller 65 may be a web-specific COM component or components that may run in a process space associated with the IIS 64. These components may be operable to start, stop, and send data 69 (such as insurance claim consultation data entered in response to insurance claim consultation questions) to the rules engine 61. These components may also be operable to receive data (such as insurance claim consultation questions and elements of the user interface) from the rules engine 61 for inclusion in one or more web pages 67. Generally, these components are configured to translate data between HTML on the IIS 64 side and the interface exposed by COM components 63b on the other side. These components may include functionality such as data validation (e.g., determining if datatypes of entered data are valid). The components may also ensure that the state of the interactions or

“conversation” between a rules engine and a browser is preserved, as discussed in greater detail with respect to Figure 4b and Figure 9.

In one embodiment, the ASP controller 65 may include at least two COM components: one which handles interactions between a web browser 68 and the rules engine executable file, and another which handles interactions between the web browser 68 and a reference system or help system executable file. The reference system executable file may provide the user with detailed assistance in conducting an insurance claim consultation session.

In one embodiment, the COM component(s) for accessing the reference system may include methods including, but not limited to, the following: addedRefsysID, initializeContentsGraphs, startSessionIfNecessary, MemberOftrueHierarchyIds, lastSearchText, lastSelectedChapterObjectId, terminateSession, getFirstMessage, pageHasError, getListParameter, chapterWasSelected, writeRefsysContents, writeContextContents, writeSearchResults, writeHelpTextAsHTML, contextHelpWasSelected, isSessionStarted, searchHitWasSelected, mergeLostBoys, searchWasSelected, and iisSessionId. Appropriate parameters may be defined for each method.

In one embodiment, the COM component(s) for accessing the rules engine 61 may include methods including, but not limited to, the following: terminateSession, startSessionIfNecessary, writePredisplayHtml, handleExitProcessing, getFirstMessage, pageToShow, errorMessage, pageHasError, pageWasPosted, doPageTransaction, getSingleParameter, getListParameter, getListParameterNoTrim, debugIt, formatAdsDate, hasSaveButton, hasBackButton, hasNextButton, hasContentsButton, hasCommentsButton, hasUnknownButton, hasReportButton, claimKeyFormat, statusMessage, iisSessionId, and isSessionStarted. Appropriate parameters may be defined for each method.

Figure 6: Browser-based User Interface

Figure 6 illustrates an example of a browser-based user interface for the insurance claims processing system according to one embodiment. The browser window 100 may be displayed in a display device 50b coupled to a client computer system. Typically, a web browser includes a set of standard navigation commands. As shown in Figure 6, examples of these commands may include “back” 110 to move to the previously visited page, “forward” 112 to move to the page previously visited before selecting “back,” “reload” 114 to obtain and redisplay the current page from the server, and “home” 116 to move to a previously designated home page. These standard navigation commands may be made available to the user as menu items and/or as buttons or other GUI elements. A button may be “pushed,” often by a mouse click or appropriate keyboard key, to initiate the command supplied by the button.

The browser page 104 may include an active server page or other HTML-encoded page supplied by the web server 64. The page 104 may include one or more specialized navigation commands. In one embodiment, these specialized navigation commands may be displayed as buttons or other GUI elements. In one embodiment, the specialized navigation commands may include, for example, “save” 120 to save the status of a consultation session, “help” 122 to access a reference system for insurance claim processing, “exit” 124 to safely exit the insurance claim consultation session, “back” 130 to safely move to a previous page of the insurance claim consultation session, and “reset” 132 to reset the proper state of the browser page 104. The reset command is further described with reference to Figure 9.

Insurance claim assessment data and/or insurance claim assessment questions 140 may also be displayed in the browser page 104. For example, for a given step in the insurance claim consultation session, one or more questions may be asked regarding bodily injuries and/or treatments thereof. A set of acceptable answers (i.e., insurance claim assessment data) may be supplied to the user, such as with a menu or series of check boxes. The user may then select from the possible answers and enter the insurance

claim assessment data. The set of acceptable answers may be dynamically generated by the rules engine based upon answers to previous questions.

Figure 7: A Method of Developing a Web-Based Insurance Claims Processing System

Figure 7 is a flowchart illustrating a method of developing a web-based insurance claims processing system according to one embodiment. The steps shown in Figure 7 may be performed in various orders according to various embodiments. In step 200, a rules engine may be developed or otherwise provided. As discussed with reference to Figure 3a, the rules engine may be configured to estimate a value of an insurance claim as a function of insurance claim assessment data entered by a user in response to insurance claim assessment questions.

In step 202, the rules engine may be wrapped with a component interface in accordance with a component architecture specification. Component interfaces are discussed in greater detail with reference to Figures 4 and 5. The component interface may include one or more definitions of methods of communication or other access to the rules engine, such as by a web server. The component architecture specification may include a Component Object Model (COM) specification.

In step 204, a web server may be provided, wherein the web server which is configured to generate a plurality of web pages which are viewable by a web browser. The methods of communication in the component interfaces may be operable to transmit the insurance claim assessment data from the web server to the rules engine and operable to transmit the insurance claim assessment questions from the rules engine to the web server.

Figure 8: A Method of Hosting a Web-Based Insurance Claims Processing System

Figure 8 is a flowchart illustrating a method of hosting a web-based insurance claims processing server with various pricing models according to one embodiment. In

step 250, an insurance claim processing server may be hosted. As used herein, "hosting" may include installing, maintaining, and/or otherwise providing client access to a server. The insurance claim processing server may be configured to estimate a value of an insurance claim as a function of insurance claim assessment data entered by a user during an insurance claim consultation session. In one embodiment, the insurance claim processing server may include a rules engine and a web server, and the client software may include a web browser. The web server may be operable to generate web pages and receive responses and requests from the web browser to enable communication between the rules engine and the web browser.

In step 252, client software such as a web browser may be provided to a user such as an insurance company. In one embodiment, the client software may include commercial, off-the-shelf web browser software which may already be in use by an insurance company and its employees who seek to access to the insurance claim processing server. The client software may be operable to receive the insurance claim assessment data entered by the user and send the insurance claim assessment data across a network to the insurance claim processing server. The insurance claim processing server may be operable to send the estimate of the value of the insurance claim to the client software across the network. In one embodiment, the network may include the Internet.

In step 254, the user may be charged for access to the insurance claim processing server through client software according to a pricing model. Various pricing models may be used with various embodiments of the hosting system and method. The pricing model may include a fee for each of a plurality of insurance claim consultation sessions conducted by the user. The pricing model may include a fee for each fixed period of access time of access by the user to the insurance claim processing server through the client software. For example, the fixed period of access time may include an hourly multiple, a weekly multiple, a monthly multiple, a yearly multiple, or a multiple of minutes. The pricing model may include a fee which varies directly with an amount of time spent accessing the insurance claim consultation session through the client software.

The user may include an insurance organization having a particular size, and the pricing model varies according to the size of the user. The size of the user may include a function of a quantity of employees of the user, a function of a revenue of the user over a period of time, and/or a function of a quantity of consultation sessions conducted by the user over a period of time. The pricing model may include a pricing discount given to the user after a particular quantity of insurance claim consultation sessions conducted by the user in a particular period of time. The insurance claim consultation session may include one or more insurance claim consultation transactions, and the pricing model may include a fee for each of a plurality of insurance claim consultation transactions conducted by the user during one or more insurance claim consultation sessions.

The method may further include charging additional users for access to the insurance claim processing server through client software according to a same or different pricing model.

Figure 9: A Method of Using a Reset Button for a Web-Based Insurance Claims Processing System

Figure 9 is a flowchart illustrating a method of using a reset button provided by a web-based interface to a web-based insurance claims processing server according to one embodiment. In step 302, a first page of insurance claim assessment data may be displayed in a browser program executing on a computer system. The browser program may include a web browser program which is operable to read and display web pages. The computer system which executes the browser program may include a client computer system which is communicatively coupled to a server computer system. The server computer system may be operable to generate and send a plurality of pages of insurance claim assessment data to the client computer system.

In one embodiment, in step 304, one of the specialized navigation commands, such as a forward command, may be selected to advance to a second page of insurance

claim assessment data. In another embodiment, the user may advance to the second page by hitting "return" or otherwise instructing the insurance claim processing server to provide a next page in a consultation session. In step 306, the second page of insurance claim assessment data, including the specialized navigation commands, may be displayed in the browser.

In step 308, after the second page of insurance claim assessment data is displayed, one of the standard navigation commands, such as the "back" command or button available in a toolbar or menu in a web browser, may be selected to move back to the first page of insurance claim assessment data. The first page of insurance claim assessment data may then be redisplayed.

In step 310, the user may attempt to perform an insurance claim assessment task on the redisplayed first page of insurance claim assessment data. For example, the user may attempt to save a status of an insurance claim consultation by pressing a "save" button in the specialized buttons. The insurance claim consultation may include an interactive determination of an estimate of a value of an insurance claim through the entry of insurance claim assessment data in response to insurance claim assessment questions. The insurance claim assessment task may include selecting one of the other specialized navigation buttons provided as the user interface by insurance claim processing server. The insurance claim assessment task may also include entering new or modifying existing insurance claim assessment data. Insurance claim assessment data may include information relevant to an estimate of a value of an insurance claim, such as bodily injuries and treatments thereof. The insurance claim assessment data may include bodily injury claim assessment data, and the insurance claim assessment task may include a bodily injury claim assessment task.

In one embodiment, the state of the "conversation" between the browser and the insurance claim processing server may be preserved by a COM component 66, as discussed with reference to Figure 5. In step 312, therefore, a navigation error may be generated as a result of the attempting to perform an insurance claim assessment task on

the first page, when the second page is the “correct” page in the conversation. In one embodiment, a navigation error message may be generated and displayed to the user as a result of the generating the navigation error. The navigation error message may include an instruction to select a reset command, wherein the reset command is one of the specialized navigation commands.

In step 314, the user may select the reset command after viewing the navigation error message. In one embodiment, the insurance claim processing server may automatically perform a reset function without user intervention as a result of the navigation error.

In step 316, the second page (i.e., the “correct” page) of insurance claim assessment data may then be redisplayed. The user may then perform a second insurance claim assessment task on the redisplayed second page of insurance claim assessment data.

Although the system and method of the present invention have been described in connection with several embodiments, the invention is not intended to be limited to the specific forms set forth herein, but on the contrary, it is intended to cover such alternatives, modifications, and equivalents as can be reasonably included within the spirit and scope of the invention as defined by the appended claims.